



Preserving Area Coverage in Sensor Networks with a Realistic Physical Layer

Antoine Gallais, François Ingelrest, Jean Carle, David Simplot-Ryl

► To cite this version:

Antoine Gallais, François Ingelrest, Jean Carle, David Simplot-Ryl. Preserving Area Coverage in Sensor Networks with a Realistic Physical Layer. 26th Annual IEEE Conference on Computer Communications (INFOCOM 2007), May 2007, Anchorage, United States. inria-00136681

HAL Id: inria-00136681

<https://inria.hal.science/inria-00136681>

Submitted on 15 Feb 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Preserving Area Coverage in Sensor Networks with a Realistic Physical Layer

Antoine Gallais, François Ingelrest, Jean Carle and David Simplot-Ryl

IRCICA/LIFL, University of Lille 1.

CNRS UMR 8022, INRIA Futurs, France.

Email: {Antoine.Gallais, Francois.Ingelrest, Jean.Carle, David.Simplot}@lifl.fr

Abstract—We consider the problem of activity scheduling and area coverage in sensor networks, and especially focus on problems that arise when using a more realistic physical layer. Indeed, most of the previous work in this area has been studied within an ideal environment, where messages are always correctly received. In this paper, we argue that protocols developed with such an assumption can hardly provide satisfying results in a more realistic world. To show this, we replace the classic unit disk graph model by the lognormal shadowing one. The results show that either the resulting area coverage is not sufficient or the percentage of active nodes is very high. We thus present an original method, where a node decides to turn off when there exists in its vicinity a sufficiently reliable covering set of neighbors. We show that our solution is very efficient as it preserves area coverage while minimizing the quantity of active nodes.

I. INTRODUCTION

A. Context

Wireless sensor network (WSN) nodes are small energy-constrained devices, which are able to gather various information about their environment thanks to a monitoring module. These devices may also communicate with each other by using radio transmissions. Randomly deployed over a given area, they form a wireless network. Collected data is forwarded to a base station where further heavy computation and analysis tasks may be achieved. Typical uses of such networks include military surveillance, habitat and environmental monitoring.

To increase the network lifetime, activity scheduling is considered as a solution of prime importance: it aims at switching devices from an active (and energy expensive) state to a suspended one. As WSN are decentralized, these decisions must be made *locally* (i.e., based on neighborhood information) to avoid communication overhead. Moreover, such approach is very scalable against the size and the density of the network.

However, activity scheduling must not disturb the monitoring task of the WSN. This means that, regardless of the global state of sensors, the area on which they were deployed must be completely monitored (*covered*). Most of the known solutions consider an ideal communication model, where signal strength attenuation is not considered. Thus, no message ever gets lost.

B. Contribution

Because of the radio channel randomness, neighborhood information may get corrupted, leading to incoherent activity

scheduling. This behavior is generally not considered because of the use of an ideal signal propagation model. In this paper, we first demonstrate that the use of a realistic environment, where correct message receptions depend on link quality, highly impacts the performance of existing activity scheduling solutions. We indeed show that either the percentage of covered area is insufficient or the number of active nodes significantly increases. We then provide an original scheduling solution tolerant of message loss, based on link reliability. We show that it is very efficient compared to existing ones, both in terms of covered area and active sensors.

The remainder of this paper is as follows. We first provide in Sec. II an overview of existing work, and then we give our models and assumptions in Sec. III. In Sec. IV are provided the performances of some existing solutions when a realistic environment is considered, while our solution is described and analyzed in Sec. V. We finally conclude in Sec. VI.

II. RELATED WORK

As stated before, we focus on localized methods that use only local information. Such a simple algorithm, named PEAS, was proposed in [1], where asynchronized networks are considered. At first, all nodes are turned off. Then, periodically, each sleeping node awakes and probes its neighborhood, within a configurable range. If no active node replies, the node decides to keep its active status until running out of energy. Otherwise, the node goes back to another sleeping period. The behavior of this algorithm highly depends on the ratio between the sensing and the communication radii, which may be difficult to determine. Moreover, the connectivity of the resulting set was not studied in [1].

Zhang and Hou [2] described an algorithm for selecting covering sensors in a synchronized network. In each round, a single sensor starts the decision process, which then propagates to the whole network. New sensors are selected so that the priority is given to sensors located near to vertices in a regular hexagonal tiling. The need for a single sensor to start the process may cause problems in applying it, including increased latency. Another problem is that the original sensing area coverage is not preserved, as shown by experimental results.

An activity scheduling mechanism, providing a global coverage, is proposed in [3]. After a neighbor discovery phase, each sensor waits for a random timeout and computes the

coverage offered by its neighbors. If its own sensing area is completely covered by others, it decides to be passive and sends a withdrawal message to its neighborhood. Thus, neighbors that have not yet made their decision may update their area coverage. Otherwise, it decides to be active, and no message is sent. Results provided for the ideal signal propagation model show that this algorithm outperforms PEAS as it greatly reduces the percentage of active nodes while preserving a complete coverage. We especially focused on this algorithm since both the assumptions it relies on and the results it provides illustrate the typical area coverage protocol.

In [4], a low communication overhead protocol was proposed. Initially, each node is active and waits for a random timeout to advertise its status. Several variants have been proposed depending on the nature of the announcement. In one of these variants, called *positive only*, if a node is covered by all of its known neighbors, it decides to be passive without sending any message. If the provided coverage is not complete, it decides to be active and sends a positive acknowledgment. This method was shown to be very resistant to message loss. Furthermore, no a priori neighbor information is required since knowledge comes with activity messages themselves.

III. PRELIMINARIES

A. Network model

We model a wireless network by a graph $G = (V, E)$, where V is the set of vertices (the sensors, or the *nodes*) and $E \subseteq V^2$ the set of edges that gives the available communications: there exists an ordered pair $(u, v) \in E$ if the node u is physically able to send a message to v . The neighborhood set $N(u)$ of a node u is defined as:

$$N(u) = \{v \in V \mid v \neq u \wedge (u, v) \in E\}. \quad (1)$$

Each sensor has a communication range R_c and a sensing range R_s . We denote by $S(u)$ the area covered by a node u and $S(A)$ the area covered by a set of nodes $A = \{a_1, a_2, \dots, a_n\}$ such that:

$$S(A) = \bigcup_{i=1}^{|A|} S(a_i). \quad (2)$$

B. Radio channel model

Given a graph $G = (V, E)$ and a communication range R_c , the *unit disk graph* (UDG) model defines the set of edges E as:

$$E = \{(u, v) \in V^2 \mid u \neq v \wedge \text{dist}(u, v) \leq R_c\}, \quad (3)$$

$\text{dist}(u, v)$ being the Euclidean distance between nodes u and v . This model assumes that all links are equally reliable, which is not a realistic assumption. Indeed, due to the nature of radio waves, the signal strength greatly decreases with the distance. This behavior is generally ignored, while it may greatly impact transmissions because of the errors it generates.

Correction codes may be used to correct them if their rate is sufficiently low, but this highly depends on the considered

environment. Re-emission of corrupted messages supposes the existence of an acknowledgment process that may be unavailable. We thus assumed that no special mechanism is available: we chose to locate our work at the application layer, based only on link reliability. This is highly relevant to this research topic, because assuming that packet loss is taken care at a lower layer could be very costly.

The reliability of a link is influenced by a lot of factors such as its length, the emitting power, or the existence of obstacles. To model the reliability, we chose to use the *lognormal shadowing* model, described by Quin and Kunz in [5]. G is thus transformed into a weighted graph, where the weight of each edge $(u, v) \in E$ is equal to the probability of correct reception $p(\text{dist}(u, v))$ for the two nodes u and v . In our simulations, we used an approximated function $P(x)$, described by Kuruvila et al. in [6] as follows:

$$P(x) = \begin{cases} 1 - \frac{(\frac{x}{R_c})^{2\alpha}}{2} & \text{if } 0 < x \leq R_c, \\ \frac{(\frac{2R_c - x}{R_c})^{2\alpha}}{2} & \text{if } R_c < x \leq 2 \times R_c, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

In this formula, α is the power attenuation factor which depends on the environment, x is the considered distance and R_c is the theoretical communication range of a node. This function assumes that the probability of correct reception for the range R_c is always equal to $P(R_c) = 0.5$. Fig. 1 illustrates this function for $\alpha = 2$, which is a value frequently encountered in the literature. For the sake of clarity, $p(u, v)$ is equivalent to $P(\text{dist}(u, v))$.

C. Assumptions

We assume that a given node u is able to determine the value of $p(v, u)$ for any neighbor v . In a practical context, acquiring such a knowledge may be done by determining the signal-to-noise ratio of previous transmissions. Another method may consist in sending a given number of beacon messages and then in counting how many of them were correctly received. We also assume that sensors are sufficiently synchronized,

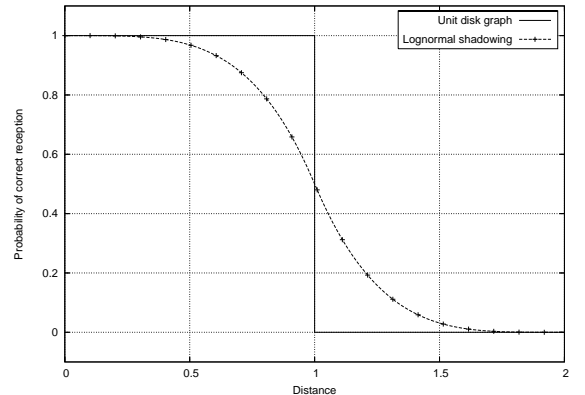


Fig. 1. The two considered physical models ($R_c = 1, \alpha = 2$).

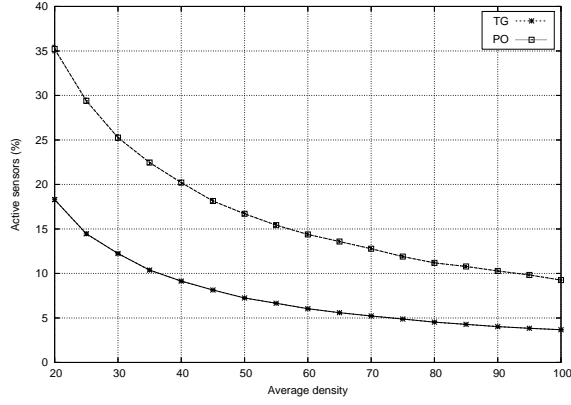


Fig. 2. Percentage of active nodes (UDG model).

problem studied in [7], in order to organize decisions into rounds and that sensors are able to compute their location [8], [9].

IV. IMPACT OF A REALISTIC PHYSICAL LAYER

Results presented in this section were obtained thanks to a home-made simulation tool. In order to determine the knowledge of the sensors, a beacon message is sent by each node u at the start of each measure, this message being affected by the considered physical layer. Then, each neighbor v of u which receives this message adds u to its neighborhood table.

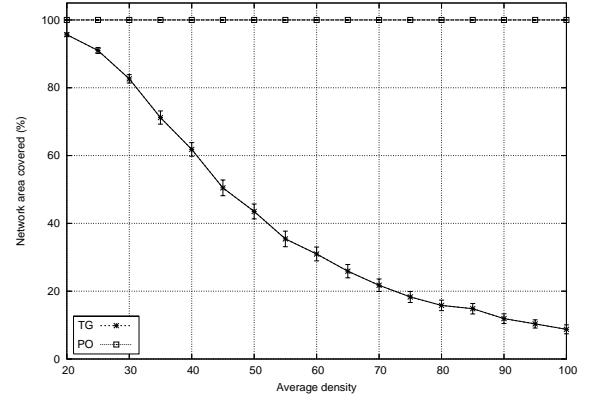
In our simulations, the WSN was deployed over a square area S of size 6×6 . Sensors all have a communication range $R_c = R_s = 1$. To deploy nodes, we considered a Poisson point process in S of intensity $\lambda > 0$. We define the density d of the network to be the average number of nodes in a given communication area: we thus have $d = \lambda \times \pi R_c^2 = \lambda\pi$. Each measure was obtained by averaging 100 iterations, and for each of the latter a new network was generated. On each figure is given the 95% confidence interval. Evaluation of the covered area was done by discretizing the network area and by observing each point thus obtained.

We chose to focus on existing protocols with no major assumptions (e.g., no centralization, no hexagonal tiling). We thus selected:

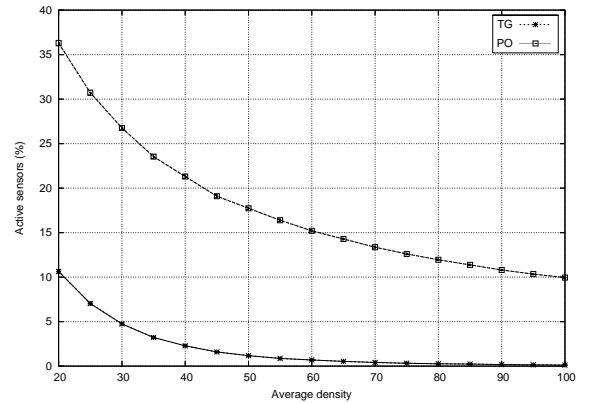
- [3], referred to as TG.
- [4], referred to as PO, because it is based on the use of *positive* messages which may be lost without impacting area coverage.

Both of them provide a total coverage of the environment when no packet loss occur. We did not consider PEAS and its variants because they are also based on positive messages, just like PO.

We provide in Fig. 2 the performance of the selected protocols within the ideal environment of the unit disk graph. One can obviously observe that TG keeps far less sensors active than PO to provide a total coverage. These results are coherent with [4]. The percentage of active sensors is kept very low and demonstrates the efficiency of the mechanism. For the



(a) Covered area



(b) Active nodes

Fig. 3. Performance of the considered protocols with the lognormal shadowing model ($\alpha = 2$).

density $d = 20$, only 18% of the sensors are active while ensuring a complete coverage. This percentage varies with the density: 6% of activity is needed for $d = 60$ and only 3.6% for $d = 100$.

However, performance of this protocol greatly decreases when a more realistic model is used, as shown on Fig. 3. Thus, with the lognormal shadowing model ($\alpha = 2$), the coverage is nearly complete only for very small densities, and is very low for higher ones. Only 31% of the environment is covered when $d = 60$ against only 9% when $d = 100$.

This performance decrease is obviously caused by the loss of withdrawal messages: when a node u decides to switch to sleep mode and alerts its neighborhood, the sent message may not be correctly received by all its neighbors. Consequently, the latter may falsely believe that u is still active and may then decide to turn off. The global coverage of the network thus decreases. This is obvious when observing Fig. 3(b), where a larger part of the sensors are asleep. This phenomenon is less visible in sparser networks, because the probability that a node finds a covering set of neighbors is lower. The decision to become passive is thus taken less frequently.

The protocol PO, by using positive messages, is protected against these packet loss and ensures a complete coverage within the realistic environment. However, this is done to the detriment of energy savings: intuitively, as messages get lost, more nodes become active. This behavior is clearly illustrated by Fig. 3(b).

V. OUR ACTIVITY SCHEDULING METHOD

The most straightforward solution, using positive messages, was shown in the previous section to preserve area coverage to the detriment of energy savings. We present in this section an original method to minimize energy consumption while preserving area coverage. Our solution works correctly in both ideal and realistic physical environments.

A. Description of our solution

Our method is based on TG, and is denoted by TGim. In this section, for a node u , *active neighbors* denote the neighbors that are supposed to be active, i.e., the neighbors from which u did not receive any withdrawal message. Our solution introduces the concept of confidence in the active neighbors: the principle, for a node u , is to determine the risk run when granting its confidence to its active neighbors. When u makes its activity decision, three cases may happen:

- 1) The set of active neighbors does not cover the sensing area of u : the latter decides to be active.
- 2) Active neighbors form a covering set, but the risk is too high: u decides to be active.
- 3) Active neighbors form a covering set, and the risk is sufficiently low: u decides to turn off.

For a given node u , the risk level may be expressed as: ‘among all my active neighbors, what is the probability that at least one of them sent me a withdrawal message that I did not receive’. If this probability is high, then the probability that the area will not be covered once u asleep is also high. If we denote by $A_u = \{a_1, a_2, \dots, a_n\}$ the set of active neighbors of u , then the risk is equal to:

$$\text{Risk}(A_u) = 1 - \prod_{i=1}^{i=|A_u|} p(a_i, u). \quad (5)$$

This formula is based on probabilistic logic: the probability that *at least* one event did not occur is the opposite of *all* of them occurred. We now suppose that each node computes the risk level of its active neighbors and compares it with a given threshold. Beyond the latter, the risk is too important to grant its confidence to its neighbors: the decision to be active is thus taken.

B. Limiting the risk level

In order to correctly approximate the risk level, it is essential for a node to not consider all its active neighbors. Indeed, not all of them are needed to form a covering set, and the risk may thus artificially increase.

A first and straightforward solution could consist in removing *weak* neighbors, for which the probability of correct

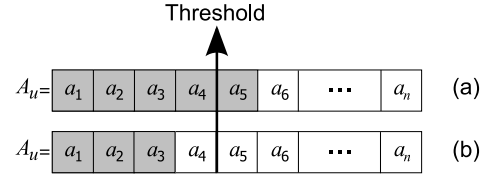


Fig. 4. Only removing weak neighbors may lead to overestimate the risk.

reception is too low. In other words, a node would immediately remove from its neighborhood table all the neighbors with a probability lower than a given threshold, and would thus only consider *strong* neighbors.

However, such a simple solution still artificially increases the risk level, as illustrated by Fig. 4. In the latter, A_u is supposed to be sorted by descending probabilities, such that $p(a_1, u) \geq p(a_2, u) \geq \dots \geq p(a_n, u)$, and grey cells represent the mandatory neighbors to form a covering set. One can clearly see in case (b) that node a_4 is going to be considered for the computation of the risk because $p(a_4, u)$ is high enough, while it is not needed to cover the sensing area of u .

A better solution consists in determining a covering subset composed of the most reliable neighbors, and then to compute the risk run by trusting this subset. In the worst case, the risk will be too high and the result will be equivalent to Fig. 4(a). In most cases however, the result will be better (refer to Fig. 4(b)).

Intuitively, finding such an optimal set is a NP-complete problem. In order to find an approximated solution, we use a simple greedy heuristic, referred to as *BestSubset*. Given a node u , a list A_u of active neighbors sorted in descending order of probabilities, and an empty set B_u , it may be described as follows. While A_u is not empty and $S(u) \not\subseteq S(B_u)$, remove the leading element from A_u and add it to B_u . This algorithm thus constructs a covering set B_u composed of the most reliable active neighbors of u . The final decision process, applied at each round, is described by the algorithm 1.

C. Using a risk threshold

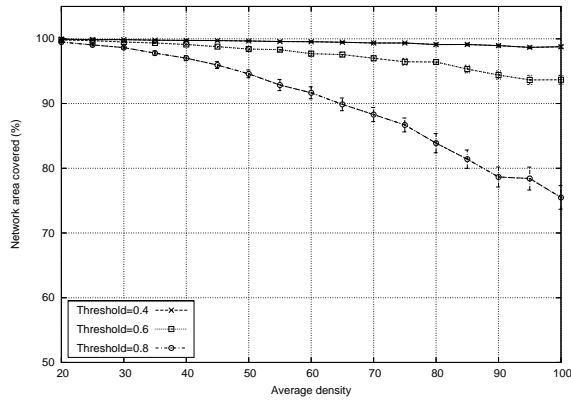
We consider in this section that each node has a static threshold, which is used to determine how high the risk may be. The results we provide here were obtained with the same parameters as in Sec. IV. We considered three static thresholds: 0.4, 0.6 and 0.8.

In Fig. 5(a) is given the percentage of covered area provided by TGim with the lognormal shadowing model. One can observe that in the three cases, the percentage of covered area is far better than the one obtained with the original protocol TG. With a threshold equal to 0.8, for which sensors run high risks, the coverage is very good for low densities. A threshold equal to 0.6 provides an excellent coverage level for all densities, as it is always over 92%. Such high thresholds may be used while preserving area coverage, because only the most reliable active neighbors are considered when evaluating the risk, which is thus generally low.

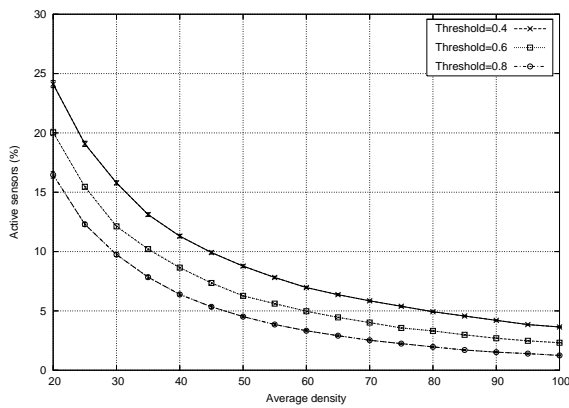
Algorithm 1: Activity decision for a node u

Input: A list N of neighbors known by u
Output: True if u is active, false otherwise
Data: A and B : lists of nodes, timeout: duration

```
A ← N
timeout ← Randomly chosen timeout
while current time < timeout do
    if reception of a message from a neighbor  $v$  then
        A ← A \ { $v$ }
    end
end
B ← BestSubset(A)
if B covering set and Risk(B) ≤ threshold then
    SendWithdrawalMsg()
    return false
else
    return true
end
```



(a) Covered area



(b) Active nodes

Fig. 5. Performance of TGim with the lognormal shadowing model ($\alpha = 2$).

The corresponding percentage of active nodes is given in Fig. 5(b). As expected, this value varies with the considered threshold. One can observe that even with a threshold equal to 0.4, which provides more than 98% of covered area for all densities, the percentage of active sensors is always lower than the one obtained with PO in the same environment.

VI. CONCLUSION

From the variety of results presented, we saw that a realistic signal propagation model leads either to incoherent or too expensive activity scheduling. We especially focused on the protocol proposed by Tian and Georganas, but we foresee that our results may be expanded to most other existing solutions. We also proposed a fault-tolerant and efficient solution based on link reliability, which may be used whenever a realistic physical layer is considered. It is also interesting to note that, with an ideal model, our method behaves just like the protocol by Tian and Georganas.

As future work, we would like to improve our solution by using a dynamic computation of the risk threshold, so that there would be no need for tuning depending on the network density. We are also investigating the impact of a non ideal MAC layer, where message loss would not rely only on the distance between the emitter and the receiver.

ACKNOWLEDGMENTS

This work was partially supported by a grant from CPER Nord-Pas de Calais/FEDER TAC COM'DOM, INRIA research action IRAMUS and CNRS National platform RECAP.

REFERENCES

- [1] F. Ye, G. Zhong, J. Cheng, S. Lu, and L. Zhang, "PEAS: A robust energy conserving protocol for long-lived sensor networks," in *Proceedings of the IEEE International Conference on Distributed Computing Systems (ICDCS)*, Washington, USA, May 2003.
- [2] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc and Sensor Wireless Networks, an International Journal*, vol. 1, pp. 89–123, 2005.
- [3] D. Tian and N. Georganas, "A coverage-preserving node scheduling scheme for large wireless sensor networks," in *Proceedings of the ACM International Workshop on Wireless Sensor Networks and Applications*, Atlanta, USA, 2002.
- [4] A. Gallais, J. Carle, D. Simplot-Ryl, and I. Stojmenović, "Localized sensor area coverage with low communication overhead," in *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom)*, Pisa, Italy, March 2006.
- [5] L. Quin and T. Kunz, "On-demand routing in MANETs: The impact of a realistic physical layer model," in *Proceedings of the International Conference on Ad-Hoc, Mobile, and Wireless Networks*, Montreal, Canada, 2003.
- [6] J. Kuruvila, A. Nayak, and I. Stojmenović, "Hop count optimal position based packet routing algorithms for ad hoc wireless networks with a realistic physical layer," in *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, Fort Lauderdale, USA, 2004.
- [7] K. Romer, P. Blum, and L. Meier, *Handbook of Sensor Networks: Algorithms and Architectures* (I. Stojmenović, ed.), 2005, ch. Time Synchronization and Calibration in Wireless Sensor Networks, pp. 199–238.
- [8] L. Doherty, K. Pister, and L. Ghaoui, "Convex position estimation in wireless sensor networks," in *Proceedings of the IEEE INFOCOM*, Anchorage, Alaska, April 2001.
- [9] D. Niculescu and B. Nath, "Ad hoc positioning system (APS) using AoA," in *Proceedings of the IEEE INFOCOM*, San Francisco, USA, April 2003.